IOSR Journal of Engineering (IOSRJEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
Vol. 10, Issue 1, January 2020, ||Series -I|| PP 66-71

Online Toxic Comments Classification using Machine Learning Algorithms

Panchareddy Gayathri[1], Raghuram Naidu Challa[2]
[1,2]Assistant Professor in Department of Computer Science and Engineering (CSE), Sanketika Institute of Technology and Management (SITAM), Visakhapatnam, AP.

ABSTRACT
Due to the recent events of mass spreading of COVID-19, every physical operation has been hard stuck with its effect and has caused a sudden exponential increase in the usage of internet services. With everything going online and people using it for their respective activities, there has been a widespread of cyber bullies causing negative online behaviors, including comments that are rude, disrespectful or otherwise likely to make someone leave a discussion. The paper addresses the problem of classifying toxic comments into subcategories to help the online moderation. This paper has employed Multi-Label Classification using Problem Transformation methods.

Keywords: Multi-Label Classification, Binary Relevance, Classifier Chain, Machine Learning Algorithms.

I. INTRODUCTION

Toxic comments are defined as comments that are rude, disrespectful, or that tend to force users to leave the discussion. If these toxic comments can be automatically identified, we could have safer discussions on various social networks, news portals, or online forums. Manual moderation of comments is costly, ineffective, and sometimes infeasible. Automatic or semi-automatic detection of toxic comments is done by using different machine learning methods, mostly different deep neural networks architectures.

Detecting Toxic comments has been a great challenge for all the scholars in the field of research and development. This domain has drawn a lot of interest not just because of the spread of hate but also people refraining people from participating in online forums which diversely affects all the creators/content-providers to provide a relief to engage in a healthy public interaction which can be accessed by the public without any hesitation. The study by Ojasvi Jain, Muskan Gupta, Sidh Satam, Siba Panda[1] estimated that 30.30% of the people under survey were negatively impacted by it and 37.88% of them, even though not impacted, were still the victims of it and merely 4.55% of the victims took legal action against the bully by reporting them to the concerned government authorities. These studies and results related to this kind of online harassment leads to an important area of data science in order to be able to separate and distinguish harassment comments and cyberbullying, we call them toxic comments, from normal comments. In this paper, we address this problem of correctly categorizing toxic comments into their types. The comment data was gathered from Kaggle[2], which includes Wikipedia comments. We used Binary Relevance and Classifier Chain methods as our Problem Transformation to classify comments into toxic, obscene, insult, severe-toxic, identity-hate and threat as the labels. Data was pre-processed and features were selected and extracted. These features were used to train and test a number of candidate machine learning classifiers: Logistic Regression, Gaussian Naive Bayes Classifier, Multinomial Naive Bayes Classifier , Decision Tree Classifier.

II. RELATED WORKS

There have been several studies regarding toxic comment classification in the past few years. The authors of the paper[3] implemented a Convolutional Neural Network (CNN) with character level embedding for detecting types of toxicity in online comments, obtaining a Mean Accuracy of 94%. Another study by Kevin Khieu and Neha Narwal[4] studies the impact of SVM and Neural networks like CNN, LSTM, RNN on identifying toxicity in text, using multiple word embedding techniques. They conducted word-level analysis and character- level analysis using these models and concluded LSTM and CNN being the best performers among all. Almost every research conducted in the domain used the dataset released by Kaggle during the Toxic Comment Classification Challenge. In this paper we extend the research based on the suggestion of the authors of [5]. We took forward the task of identifying if a comment is toxic or not (binary classification) to classifying the toxic comment into respective labels it belongs to (multi-label classification).

Machine Learning Methods for Toxic Comment Identification and Classification [3] Internet traffic has increased exponentially over the last four months as a result of the ongoing pandemic. This has resulted in a large number of enthusiastic new and old clients using the internet for a wide range of services, including academic, entertainment, industrial, and monitoring, as well as the emergence of a new tendency in business life known as work-from-home. As a consequence of the unexpected rise in the number of people using the internet, the number of cunning people has increased. Nowadays, the top goal for any internet platform provider is to keep inclusive and positive interactions. Twitter, an online media platform where users can express their views, is the best example that can be used. This platform has already received a great deal of criticism due to the proliferation of hate speech, insults, threats, and defamatory acts, making it difficult for many internet service providers to regulate them. As a result, studies into the classification of toxic comments are presently underway. On the dataset, we combine non-identical machine learning and other unimportant methods to propose a model that outperforms them all when compared side by side. For the reasons stated above, we used the Kaggle dataset, which is a well-known and valuable resource for academics attempting to understand the issue of toxic comment classification. The findings would aid in the development of an online interface that would allow us to determine the level of toxicity contained in a particular phrase or sentence and categorize it accordingly. Classification of Toxic Comments [4]. Conversational toxicity is an issue that can cause people to stop being themselves and stop seeking the opinions of others because they are afraid of being harassed or attacked. In this research, natural language processing (NLP) approaches are used to detect toxicity in writing and warn people before sending potentially harmful informational messages. Natural language processing (NLP), a subset of machine learning, allows machines to comprehend human speech. A computer is capable of comprehending, analysing, manipulating, and even creating human data language. Natural language processing (NLP) is an artificial intelligence subset that allows computers to comprehend and analyse human language rather than merely reading it. Natural language processing enables computers to understand spoken or written English and carry out tasks such as speech identification, sentiment analysis, text classification, and automatic text summarization. (NLP). Deep Learning Classification of Social Media Toxicity: Real-World UK Application Brexit [6] social media is now widely used by people to express their perspectives on a range of issues, and it is now an integral component of contemporary culture. Social media is becoming more and more necessary for the majority of people, and there have been numerous accounts of social media addiction. Twitter and other social media platforms have demonstrated how important it is to bring people from all over the globe and from various backgrounds together over time. However, they have shown negative side effects that could have a negative influence. One such unfavourable side effect is the extreme poisonousness of many social media discussions. In this study, we develop a useful model for identifying and classifying toxicity in user-generated material on social media using Transformers' Bidirectional Encoder Representations. (BERT). The BERT pre-trained model and three of its variants were enhanced using the Kaggle public dataset, a well-known labelled toxic remark dataset. (Toxic remark Classification Challenge). We also test the proposed models using two datasets gathered from Twitter during two different time periods to detect toxicity in user-generated content (tweets) using hashtags related to the UK Brexit. The findings demonstrated how well the suggested approach classified and analysed harmful tweets.

III. PROPOSED ARCHITECTURE

This section focuses on different algorithms and the various stages that are involved for the proposed Toxic comment classification system.

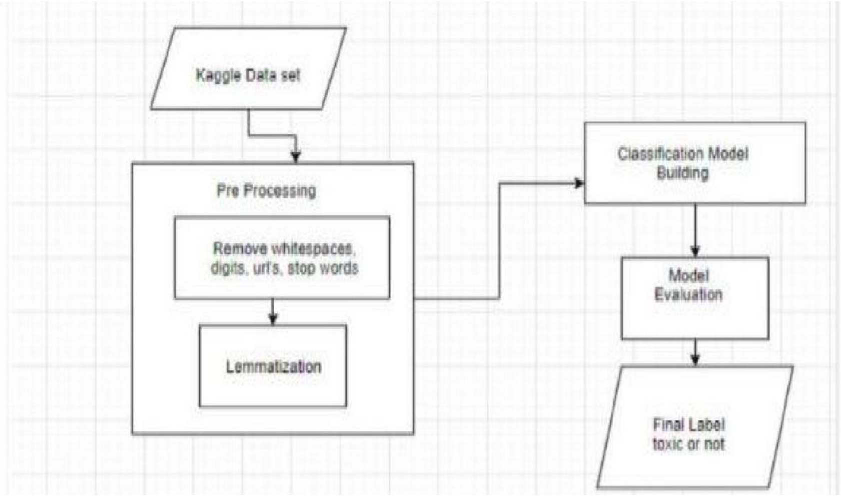


Fig 1: Proposed system

A. Exploratory Data Analysis

The dataset included around 160k comments and 6 labels to which they might belong. Fig.1 shows the distribution of labels of toxicity and Fig.2 shows the number of labels a comment belongs to (referred as Tags).

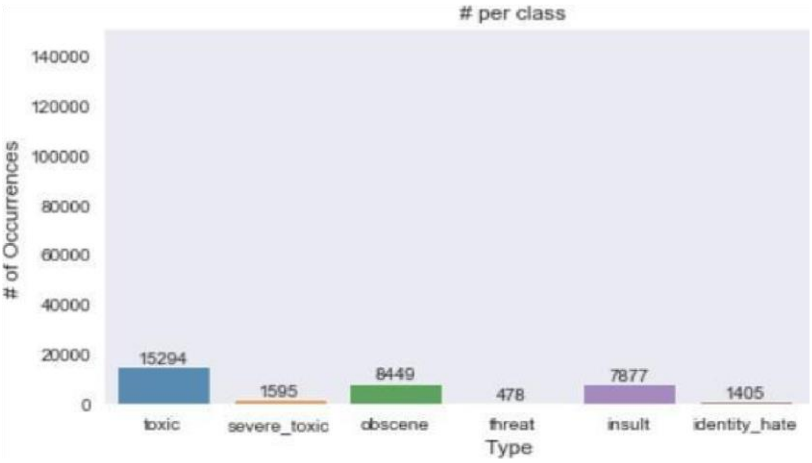


Fig 2: No. of comments per Label

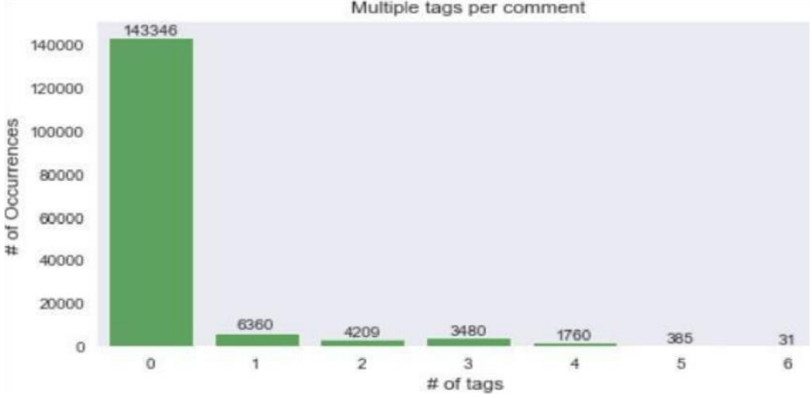


Fig 3: Tags per comment

We observed that the no. of comments present in the dataset was not consistent with the visualization we had. Upon analysis of the dataset, we found that there are some comments that do not belong to any of the labels, which is evident from Fig.3.

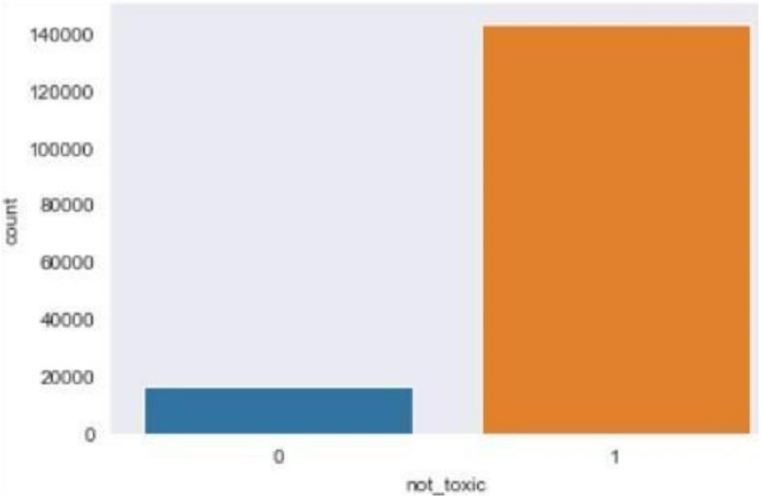


Fig 4: No. of not_toxic comments

This was a clear indication towards the facts that our dataset had a class imbalance problem. We also found that the comment text included random punctuations.

B. Data Pre-Processing

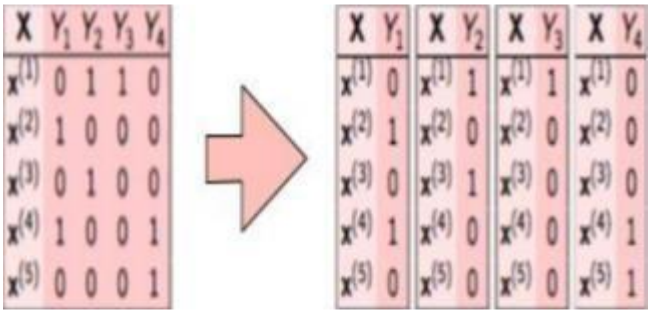
The data pre-processing for the included 2 major stages: (1) Data Cleaning; (Removal of unnecessary elements from our text); (2) Feature Engineering (extracting features from data and transforming them into formats that are suitable for Machine Learning algorithms).

For the data cleaning, the data was converted into lowercase and then punctuations and other special/ nonASCII characters were removed. We then used Stemming and Lemmatization along with removal of stopwords. To further make our vocabulary more effective we removed the most and least frequent words from the comments. The next was to convert the cleaned comments into feature vectors to make them suitable for training. We applied TF-IDF Transformation using TfidfVectorizer.

C. Multi-Label Classification Techniques

As we are tackling the problem of multi-Label classification, we need to apply Problem Transformation methods on our multi-label data to break it down to single-label problems. The conventional Machine Learning algorithms are designed for classification problems with single-label as the output, thus these transformation methods allowed us to use them for the task of Multi-Label Classification. We use the following transformation methods:

- **Binary Relevance:** The interdependence of labels is not taken into account in this process. Each label is solved separately, like a single-label classification problem. Fig.4 shows that the target labels Y1, Y2, Y3, Y4 are treated separately with respect to the input feature vector X.



X	Y_1	Y_2	Y_3	Y_4
$x^{(1)}$	0	1	1	0
$x^{(2)}$	1	0	0	0
$x^{(3)}$	0	1	0	0
$x^{(4)}$	1	0	0	1
$x^{(5)}$	0	0	0	1

X	Y_1
$x^{(1)}$	0
$x^{(2)}$	1
$x^{(3)}$	0
$x^{(4)}$	1
$x^{(5)}$	0

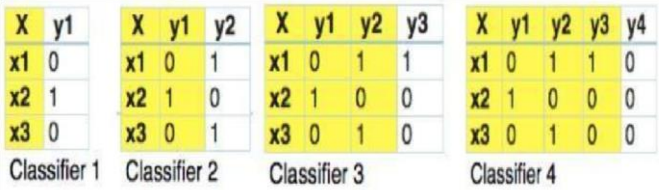
X	Y_2
$x^{(1)}$	1
$x^{(2)}$	0
$x^{(3)}$	1
$x^{(4)}$	0
$x^{(5)}$	0

X	Y_3
$x^{(1)}$	1
$x^{(2)}$	0
$x^{(3)}$	0
$x^{(4)}$	0
$x^{(5)}$	0

X	Y_4
$x^{(1)}$	0
$x^{(2)}$	0
$x^{(3)}$	0
$x^{(4)}$	1
$x^{(5)}$	1

Fig 5: Binary Relevance

- **Classifier Chain:** In this method, the first classifier is trained on the input X. Then the subsequent classifiers are trained on the input X and all previous classifiers' predictions in the chain. This method attempts to draw the signals from the correlation among preceding target variables. Fig. 5 shows that the target label becomes a part of the input feature vector as we go down the chain.



X	y1
x1	0
x2	1
x3	0

Classifier 1

X	y1	y2
x1	0	1
x2	1	0
x3	0	1

Classifier 2

X	y1	y2	y3
x1	0	1	1
x2	1	0	0
x3	0	1	0

Classifier 3

X	y1	y2	y3	y4
x1	0	1	1	0
x2	1	0	0	0
x3	0	1	0	0

Classifier 4

Fig 6: Classifier Chain

D. Training the Classifiers

The task presented itself as a supervised classification problem with output as multiple toxicity categories as labels. It was believed that given the relatively high dimensional, complex feature space, the optimum decision boundary would likely be nonlinear. Classifiers selected for evaluation were those that work well in higher dimensional space with categorical data and were capable of generating linear and nonlinear decision boundaries: Logistic Regression, Gaussian Naive Bayes Classifier, Multinomial Naive Bayes Classifier, Decision Tree Classifier. We used Sklearn library’s implementation of classifiers and implemented the Transformation Methods discussed earlier for training the models.

IV. RESULTS

The classifiers were trained and their performance was measured with Hamming-Loss, Accuracy and Log-Loss as our evaluation metrics.

Table 1: Logistic Regression

	Hamming_loss	Accuracy	Log_Loss
Binary Relevance	7.6656	66.3328	1.3559
Classifier Chain	7.6938	68.8412	1.3108

Table 2: Gaussian Naive Bayes Classifier

	Hamming_loss	Accuracy	Log_Loss
Binary Relevance	31.7873	26.8567	5.4054
Classifier Chain	31.8567	28.3204	4.8993

Table 3: Multinomial Naive Bayes Classifier

	Hamming_loss	Accuracy	Log_Loss
Binary Relevance	8.8341	62.7272	1.4171
Classifier Chain	9.1987	62.9429	1.3848

Table 4: Decision Tree Classifier

	Hamming_loss	Accuracy	Log_Loss
Binary Relevance	9.8690	57.2110	7.8113
Classifier Chain	9.9563	59.2295	7.0812

It is clearly visible that Logistic Regression bettered all other classifiers. We can also infer that among Binary Relevance and Classifier Chain, Classifier Chain outperformed the former, across different classifiers.
Multinomial Naive Bayes Classifier performed closely to Logistic Regression.

V. CONCLUSION

This paper has discussed approaches to implement various machine learning algorithms to carry out multilabel classification and compared their performance. After proper review we conclude that the Logistic Regression Classifier achieved the highest performance among the classifiers. In the future scope of our work and this paper, we plan to work on optimizing the algorithms used in the paper and use Algorithm Adaptation Methods along with more complex deep learning algorithms to improve the performance of the solution.

REFERENCES

[1]. Ojasvi Jain, Muskan Gupta, Sidh Satam, Siba Panda: “Has the COVID-19 pandemic affected the susceptibility to cyberbullying in India?”, Computers in Human Behavior Reports, Volume 2, 2020, 100029, ISSN 2451-9588.

[2]. https://www.kaggle.com/c/jigsaw-toxic-comment-classification- challenge/data

[3]. Theodora Chu, Kylie Jue and Max Wang: “Comment Abuse Classification with Deep Learning”,

[4]. https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1174/reports/2 762092.pdf

[5]. Kevin Khieu, Neha Narwal: ”Detecting and Classifying Toxic Comments”,

[6]. https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1184/reports/68 37517.pdf

[7]. Zaheri, Sara; Leath, Jeff; and Stroud, David (2020) "Toxic Comment Classification," SMU Data Science Review: Vol. 3 : No. 1 , Article 13.

Authors

PANCHAREDDY GAYATHRI., Assistant Professor in Department of Computer Science and Engineering (CSE), Sanketika Institute of Technology and Management (SITAM), Visakhapatnam, AP. A lady of true vision towards modern professional education and deep routed values. She had published her research papers in 2 international journals. She also presented papers in international and national conferences. A few more papers of her are under processing for publication. She actively participated in professional bodies at various organizations. Her areas of interest are Python Programming, Compiler Design, Object Oriented Software Engineering, Operating Systems, Machine Learning, and Database programming.

Her hobbies include listening to old and new melodies, reading books.

She believes in the wordings of **Swami Vivekananda**

"**Whatever you think that you will be. If you think yourself weak, weak you will be; if you think yourself strong, you will be**."

Mr.Raghuram Naidu Challa Working as Asst.Professor, Department Of Computer Applications in Sanketika Vidya Parishad Engineering College, Visakhapatnam-530041,Andhra Pradesh. He has More than 7 Years of Teaching Experience in Various Colleges in Andhra Pradesh. His Area of interests include Microsoft .NET, Python with Machine Learning, Data Science, Power BI, Sql Server, Oracle 12c, Data Mining and Data Warehousing and software Testing.

He believes in the wordings of Dr. **A.P.J. Abdul Kalam**

Dream is not that which you see while sleeping it is something that does not let you sleep